UNITED STATES PATENT APPLICATION

FOR

METHOD AND APPARATUS FOR DETERMINING ENFORCEMENT SECURITY DEVICES IN A
NETWORK TOPOLOGY

INVENTORS:

SHIGANG CHEN
PARTHA BHATTACHARYA
LIMAN WEI

PREPARED BY:

HICKMAN PALERMO TRUONG & BECKER LLP
1600 WILLOW STREET
SAN JOSE, CA 95125
(408) 414-1080

EXPRESS MAIL MAILING INFORMATION

"Express Mail" mailing label number   EL558817088US

Date of Deposit   November 30, 2001

METHOD AND APPARATUS FOR DETERMINING ENFORCEMENT SECURITY DEVICES

IN A NETWORK TOPOLOGY

FIELD OF THE INVENTION

[0001]   The present invention generally relates to security policies for networks. The

5   invention relates more specifically to a method and apparatus for determining enforcement

security devices in a network topology.

BACKGROUND OF THE INVENTION

[0002]   In policy-based security management, as applied in computer networks, the user

inputs a network topology and a list of end-to-end security policies to a policy manager. Each

10   security policy regulates how packets from a source network object to a destination network

object are to be treated. In particular, each security policy determines whether packets

between the source and destination nodes will be denied or permitted, and if permitted, under

what conditions. The topology describes the arrangement of network devices.

[0003]   Security management software is typically operated on a policy server that

15   enforces a security policy specified by the user. An example of security management

software is CISCO SECURE POLICY MANAGER (CSPM). The security management software

implements security policies on devices in the network, including security devices and

firewalls that may be located on the network.

[0004]   In the case where static routing is used to connect a source and destination node,

20   enforcing a security policy is simple. Static routing usually results in only one

communication path existing between the source node and the destination node. The security policy is implemented on firewalls on that communication path.

[0005] More typically, dynamic routing is used to connect the source node to the destination node. In dynamic routing, there may be numerous communication paths between the source node and destination node. In this case, the most conservative implementation of a security policy is to configure all enforcement firewalls using the policy, so that the policy is enforced on every possible routing path from the source node to the destination node.

[0006] In the case where dynamic routing is used, it is typically a challenging task to efficiently identify all enforcement firewalls that may potentially need to implement a security policy between a given source node and destination node. One current approach uses a "brute-force" methodology to find all possible paths between the source and destination nodes. Then, every path is examined to identify enforcement firewalls for the source and destination node. The "brute-force" approach is very computationally extensive, and has an exponential complexity when all paths between the source and destination nodes are computed. For example, it can take days for a policy server to determine all possible communication paths between a source node and destination node for a network having thousands of nodes.

[0007] Another approach is to use skilled technicians who can locate enforcement firewalls. The skilled technician may manually reduce redundant paths to minimize the number of firewalls in use. Because the approach is manual, it requires overhead and costs associated with employing experienced technicians. Furthermore, this kind of approach is not automated.

[0008] Based on the foregoing, there is a clear need for an efficient and automated approach for identifying enforcement firewalls that may potentially need to enforce a security policy in a network topology for a given source and destination node.

SUMMARY OF THE INVENTION

[0009] The foregoing needs, and other needs and objects that will become apparent for the following description, are achieved in the present invention, which comprises, in one aspect, a method for determining enforcement security devices in a network.

5 **[0010]** In an embodiment, a source node and a destination node are identified for a network topology. The source and destination nodes are end points for traffic that is to be sent through the network topology. Nodes in a path closure set are programmatically identified for that source node and destination node. Enforcement security devices are identified from the path closure set.

10 **[0011]** In other aspects, the invention encompasses a computer apparatus, a computer readable medium, and a carrier wave configured to carry out the foregoing steps.

50325-0626 (Seq. No. 4895)

BRIEF DESCRIPTION OF THE DRAWINGS

[0012]   The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

[0013]   FIG. 1A is a simplified example of a network topology having identifiable nodes for determining enforcement firewalls.

[0014]   FIG. 1B is another simplified example of a network topology having identifiable nodes for determining enforcement firewalls.

[0015]   FIG. 2 illustrates components for identifying enforcement firewalls and implementing a security policy on the enforcement firewalls.

[0016]   FIG. 3 is a flow chart that illustrates a method for implementing a security policy on a network using enforcement firewalls and other security devices for a given source and destination node.

[0017]   FIG. 4 is a flow chart of a method for identifying nodes in a path closure set.

[0018]   FIG. 5 is a block diagram that illustrates a computer system upon which an embodiment of the invention can be implemented.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[0019]   A method and apparatus for determining enforcement firewalls in a network is described. In the following description, for the purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the present invention. It will be apparent, however, to one skilled in the art that the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to avoid unnecessarily obscuring the present invention.

[0020]   Embodiments are described herein according to the following outline:

[0021]   1.0 OVERVIEW

[0022]   1.1 TERMINOLOGY

[0023]   1.2 GENERAL DESCRIPTION

[0024]   2.0 NETWORK DESCRIPTION

[0025]   3.0 IMPLEMENTATION ARCHITECTURE

[0026]   4.0 FUNCTIONAL DESCRIPTION

[0027]   5.0 HARDWARE OVERIVEW

[0028]   6.0 EXTENSIONS AND ALTERNATIVES

[0029]   1.0 OVERVIEW

[0030]   Embodiments of the invention provide an efficient method for determining enforcement security devices between a given source and destination node on a network topology. A security policy can then be implemented on the network using selected enforcement security devices that effectively regulate all traffic between a given source and destination node.

50325-0626 (Seq. No. 4895)

1.1 TERMINOLOGY

[0031] A "security device" is any device that can regulate traffic on a network, including denying or permitting packets of data. A security device may also include any device that will permit data packets to proceed to a destination under certain conditions. Security devices may include firewalls.

[0032] The term "enforcement security device" means a security device that affects communications between a specified source and destination node. Enforcement security devices may be configured to enforce a security policy that is specific to a source and destination.

[0033] The term "node" refers to one or more elements on a network that can be treated as having no internal topology for purposes of implementing a security policy. Examples of nodes on a network include firewalls, sub-networks ("subnets"), switches, gateways, hubs, and routers.

[0034] A "path closure set" refers to a select set of nodes between a source node and a destination node. Each node in a path closure set is part of a communication path between the source and destination. When traffic between a source node and destination node is dynamically routed, the path closure set represents a subset of nodes in the network topology that can form part of a non-looping path between the source node and destination node.

[0035] The term "adjacent node" means a node that directly communicates with another node. Adjacent nodes will exchange communications without any intermediate nodes.

[0036] The term "programmatically" means through use of programming or logic. Programmatically performed steps are automatic, and may involve one or more software elements, hardware elements, or a combination thereof.

[0037] The term "traverse" means that traffic passes from one node to another adjacent node. The term "forward traverse" means (i) a first node is forwarding traffic to an adjacent second node, and (ii) the first node did not receive the traffic from the second node before forwarding that traffic to the second node.

[0038] A "communication path" is a series of nodes that sequentially forward traverse traffic from a source node to a destination node.

[0039] A "loop path" is a path formed by forward traversing a sequence of adjacent nodes until one node is determined to repeat itself in the sequence. The sequence of adjacent nodes that form the loop path is not considered to be a communication path, but nodes in the loop path may be part of the path closure set because they are segments of other non-looping communication paths between the source and destination node.

[0040] The term "looping" means a path formed by forward traversing a sequence of adjacent nodes that is determined to contain a loop.

[0041] A "loop closure node" is a node that is repeated when a loop path is identified.

[0042] A "non-looping communication path" is a path that extends between a source and destination node, and has no "loop path" as a part of it. Nodes in a non-looping communication path may be part of looping paths that are identified in the process of identifying non-looping communication paths between a given source and destination node.

[0043] The term "link" refers to a connection between two adjacent nodes.

1.2   GENERAL DESCRIPTION

[0044] Embodiments of the invention identify nodes in a network topology that are each part of a non-looping communication path between a given source node and a destination node. Such nodes are said to belong to a path closure set. An embodiment identifies all nodes

50325-0626 (Seq. No. 4895)

in all non-looping communication paths between the given source and destination node as part of the path closure set. The enforcement security devices are identified from nodes in the path closure set.

[0045] Embodiments of the invention describe methodologies for identifying nodes in the path closure set. The steps performed in identifying nodes in the path closure set may result in locating one or more loop paths. A determination is made as to whether nodes in such loop paths are actually part of non-looping paths between the source node and destination node.

[0046] According to an embodiment, source and destination nodes are identified for traffic that is to be sent through a network topology. Select nodes in the network topology that are part of a path closure set for that source and destination node are programmatically identified. The enforcement security devices are part of the path closure set.

[0047] In an embodiment, a path closure set is identified by traversing all nodes in the network using links that connect adjacent nodes. Located nodes are designated as part of a path closure set if one of three conditions are satisfied for a sequence of adjacent nodes. A first condition states that a sequence of adjacent nodes are part of the path closure set when one of the located nodes is the destination node, and if the resulting path between the source node and the destination node is non-looping.

[0048] A second condition and third condition are for looping paths. The second condition states that a sequence of adjacent nodes are part of the path closure set when one of the located nodes is identified as a loop closure node, and if any of the nodes in the sequence other than the loop closure node are already designated as part of the path closure set. The third condition states that a sequence of adjacent nodes are part of the path closure set when one of the located nodes is identified as a loop closure node, and if a selected one of the

-9-

sequence of nodes, other than the loop closure node, is subsequently designated as part of the path closure set. The selected node may be an adjacent node to the loop closure node.

[0049] Embodiments of the invention provide several advantages over other known solutions. Among these advantages, enforcement firewalls and other security devices may be located automatically, without the need for skilled manual intervention. The need to check each path through the network topology is also avoided. Consequently, embodiments of the invention are effective as a network topology is scaled up to include many additional nodes.

[0050] According to an embodiment, the amount of time required to locate enforcement firewalls and other security devices depends only on the size of the network topology, and in particular, on the number of links between nodes in the network topology.

2.0 NETWORK DESCRIPTION

[0051] FIG. 1A is a simplified example of a network topology having identifiable nodes for determining enforcement firewalls and other security devices. A network 100 includes a plurality of nodes, including a source node 102 and a destination node 104. Other nodes in network 100 include sub-networks, routers and firewalls. As will be described, a security policy for regulating traffic between source node 102 and destination node 104 is implemented only on enforcement firewalls between the source node 102 and destination node 104.

[0052] In an embodiment, enforcement firewalls are identified from among all firewalls on the network 100. Traffic between the source node 102 and destination node 104 is regulated by the security policy implemented only on the corresponding enforcement firewalls.

-10-

[0053] Embodiments of the invention enable the enforcement firewalls on network 100 to be efficiently located, so that the security policy for regulating traffic between source node 102 and destination node 104 is implemented only on the enforcement firewalls, and not on extraneous firewalls that are part of network 100. A path closure set of nodes is identified from nodes on the network topology. The path closure set contains the nodes of all non-looping communication paths between the source node 102 and the destination node 104.

[0054] According to an embodiment, the nodes of the path closure set are identified by locating each node in the network topology, and by making a determination as to whether a path containing each located node is part of the path closure set. The individual nodes may be located by traversing the network through a series of sequentially adjacent nodes. A policy server 210 (FIG. 2) may be used to traverse among the nodes in network 100. By traversing the network 100 using adjacent nodes, policy server 210 is able to identify several paths between the source and destination nodes 102, 104. If specified conditions are met as a path is formed, nodes in that path are designated to be part of the path closure set. When the entire network topology is checked, then nodes in the path closure set corresponding to firewalls are identified as the enforcement firewalls.

[0055] One methodology for traversing network 100 is to locate nodes using a "depth-first" search. In the depth-first search, nodes are located in a sequence derived from forward traversing from one adjacent node to another adjacent node. A tentative path is constructed as a list of traversed nodes. In such a sequence, nodes may be encountered that have no other adjacent nodes to maintain the forward traversal. These nodes may be referred to as end nodes. When either an end node or destination node 104 is encountered, the traversal is backtracked until a previous node in the sequence is encountered having an adjacent node that was not previously located. The forward traversal is resumed for the sequence using the

-11-

first unchecked adjacent node encountered when backtracking from the end node or destination node 104. Nodes that were encountered during the backtrack are removed from the tentative path that was being formed from the sequence of nodes in the forward traversal. If no unchecked, adjacent nodes remain, then all nodes in network 100 have been checked, and the path is finalized.

[0056]   Nodes that are adjacent to one another are linked. Each incremental traversal across network 100 locates a node that is linked to the node that was previously located. Each located node that is linked to an adjacent, previously located node. A next node is adjacent and linked to the located node, and will be checked in the next increment. After the next increment, the next node becomes the located node. If for any located node, the next node is the previous node, then that node is an end node. For example, at the increment where the located node is a first firewall 130, the previous node is the source node 102. Depending on the direction of the forward traversal, the next node is either a first sub-network 110, or a second sub-network 112.

[0057]   In FIG. 1A, a plurality of links 128 connect adjacent nodes of network 100. Traversing from source node 102 to destination node 104 may identify several communication paths formed by a sequence of adjacent or linked nodes. From source node 102, a first path 152 may be identified that connects source node 102 and destination node 104. The first path 152 is formed by a sequence of adjacent nodes that consist of first firewall 130, first sub-network 110, a second firewall 132 and to destination node 104.

[0058]   The first path 152 satisfies a first condition for designating nodes as part of the path closure set. The first condition states that the nodes of a path formed by forward traversing across the network 100 are designated as being part of the path closure set if one of the nodes encountered during the forward traversal is the destination node 104, and if the

-12-

path formed is non-looping. Because the first condition is satisfied, the nodes of the first path 152 are identified as belonging to the path closure set.

[0059]   Once destination node 104 is identified, implementing the depth-first methodology requires policy server 210 (FIG. 2) to backtrack previously located nodes, until an unchecked

5   adjacent node is encountered. In FIG. 1A, first firewall 130 and second firewall 132 each have two adjacent nodes. Backtracking from destination node 104, second firewall 132 is identified as having an unchecked adjacent node-corresponding to fifth sub-network 118. A second path 154 may be identified by forward traversing sequential nodes from second firewall 132. Nodes encountered by backtracking from destination node 104 to second

10   firewall 132 are not considered part of the second path 154. But nodes used to forward traverse to second firewall 132 from source node 102 are part of both first path 152 and second path 154. By forward traversing from second firewall 132, the located nodes that are sequentially identified are fifth sub-network 118, first router 140, third sub-network 114, third firewall 134, second sub-network 112, first firewall 130, first sub-network 110, and

15   second firewall 132.

[0060]   When the policy server 210 (FIG. 2) encounters second firewall 132 again, it recognizes the second firewall as a loop closure node. The sequence of second firewall 132, fifth sub-network 118, router 140, third sub-network 114, third firewall 134, second sub-network 112, first firewall 130, and first sub-network 110 are a loop path. Second path 154

20   includes that loop path. Nodes in a loop path are part of the path closure set if a second condition is satisfied. The second condition states that nodes in a loop path are part of the path closure set if at least one node other than the loop closure node is already part of the path closure set. In FIG. 1A, for example, several nodes in the loop path of second path 154 are part of the path closure set, including first sub-network 110 and first firewall 130.

[0061] Another path may be identified by back tracking from second firewall 132 to first firewall 130. But second sub-network 112 has been checked, so the traversal is not performed. First router 140 also has an unchecked adjacent node, corresponding to fourth sub-node 116. But this node is an end node. Therefore, it is not part of the path closure set.

5 **[0062]** In FIG. 1A, application of the first and second conditions results in creating a path closure set including information identifying first firewall 130, second firewall 132, third firewall 134, first sub-network 110, second sub-network 112, third sub-network 114, fifth sub-network 120 and first router 140. Only fourth sub-network 116 is not part of the path closure set. There are two non-looping communication paths in network 100. The first non-
10 looping communication path corresponds to first path 152, as it extends between source node 102 and destination node 104 without looping. The second non-looping communication path may be derived from second path 154, and includes nodes identified by forward traversing from source node 102, first firewall 130, second sub-network 112, third firewall 134, third sub-network 134, first router 140, fifth sub-network 118, second firewall 132 and destination
15 node 104. Therefore, the second non-looping path communication path does not include first sub-network 110.

[0063] Firewalls and other security devices in the path closure set correspond to the enforcement security devices for source and destination nodes 102, 104. Therefore, in FIG. 1A first firewall 130, second firewall 132 and third firewall 134 are enforcement firewalls.
20 Any firewalls located past a portion of the network provided by sub-network 116 would not be considered enforcement firewalls.

[0064] In FIG. 1A, the path closure set was determined using the first two conditions because the first path 152 was identified before the second path 154. If the second path 154 was identified before first path 152, the second condition would not have yielded nodes in the

-14-

second path 154 as being part of the path closure set. Therefore, a third condition is required to determine whether nodes in a loop path are part of the path closure set when no nodes in the loop path are, at the time the loop path is identified, already designated as being part of the path closure set.

[0065]   FIG. 1B is another simplified example of a network topology having identifiable nodes for determining enforcement firewalls. In FIG. 1B, network 100 is altered from FIG. 1A by severing a link between second firewall 132 and fifth sub-network 118, and by making a new link between fifth sub-network 118 and first firewall 130. As such, fifth sub-network 118 is not adjacent to second firewall 132. Forward traversing from source node 102 may identify nodes of third path 156. The third path 156 contains second sub-network 112, third firewall 134, third sub-network 114, first router 140, and fifth sub-network 118. The first firewall 130 is its loop closure node.

[0066]   In an example provided by FIG. 1B, the set of path closure nodes is identified using a third condition. The third condition states that if a loop closure node is encountered, and the second condition does not apply, then the first node that follows the loop closure node is flagged. The loop closure node is first firewall 130. The first node that follows the loop closure node is, for a given direction, second sub-network 112. Therefore, second sub-network 112 is flagged.

[0067]   After third path 156 is identified, depth-first method is continued on network 100. From first firewall 130, first sub-network 110, second firewall 132 and destination node 104 are identified as first path 152. Since first path is a non-looping communication path between source node 102 and destination node 104, all nodes in first path 152 are part of the path closure set. After first path 152 is identified, there are no other paths that provide nodes for the path closure set under the first two conditions. Therefore, for this scenario, after all nodes

-15-

in network 100 are checked, fifth sub-network 118 is not identified as part of the path closure set. Accordingly, the path closure set for network 100 as depicted in FIG. 1B consists of nodes in first path 152. Only the loop closure node in third path 156, corresponding to first firewall 130, is a part of the path closure set, because that node is also part of first path 152. From the path closure set, it can be determined that the enforcement firewalls are first firewall 130 and second firewall 132, but not third firewall 134.

[0068] One advantage provided by the three conditions described with FIG. 1A and FIG. 1B is that the path closure set may be determined in a linear duration of time. That is, the path closure set for a network topology may be determined in a duration that is proportional to the number of links 128 between all adjacent nodes of the network 100.

3.0   IMPLEMENTATION ARCHITECTURE

[0069] FIG. 2 is a block diagram showing a system for identifying enforcement firewalls and implementing a security policy on the enforcement firewalls. The system includes a policy server 210 and a repository 230. The system also includes a terminal 220 to operate policy server 210. The policy server 210 may access network 100 (FIG. 1A and FIG. 1B) through a network channel 215.

[0070] In an embodiment, policy server 210 executes instructions to identify enforcement firewalls on network 100. This includes instructions for identifying nodes in network 100 that are part of the path closure set. As described, the path closure set may be identified by traversing a sequence of adjacent nodes sequentially using policy server 210, and determining if the sequence of adjacent nodes satisfy one of three conditions. The policy server 210 may also execute instructions for implementing security policies on the enforcement firewalls.

-16-

4.0   FUNCTIONAL DESCRIPTION

[0071]   FIG. 3 illustrates a method for implementing a security policy on a network using enforcement firewalls and other security devices for a given source and destination node. The method may be implemented on, for example, network 100 (FIG. 1A and FIG. 1B), using
5   policy server 200 (FIG. 2).

[0072]   In step 310, a source node and a destination node are identified for specified traffic across a network. The source node and destination node may be identified by a user. For example, the user of policy server 200 may specify the source node and destination node in which a security policy is to be implemented.

10   **[0073]**   In step 320, nodes belonging to the path closure set are programmatically identified. In one embodiment, each node in the network is checked, and the nodes of the path closure set correspond to a subset of all nodes in the network. Each node in the network may be checked using a computer-implemented methodology to determine if that node satisfies a criteria for being in the path closure set. The criteria may correspond to one of the
15   three conditions described with FIG. 1A and FIG. 1B. One methodology by which each node is checked is a depth-first search. Other methodologies may also be used, such as Dijkstra's algorithm and Bellman-Ford's algorithm.

[0074]   In step 330, security devices that are included in the path closure set are designated as the enforcement security devices for the specified source and destination nodes.

20   **[0075]**   In step 340, a security policy is implemented on the enforcement firewalls. In one embodiment, only the enforcement firewalls of the network are affected by the security policy. The security policy regulates traffic between the identified source and destination nodes.

-17-

[0076] FIG. 4 illustrates a more detailed method for locating enforcement firewalls and other security devices for a specified source and destination node on a network topology. The enforcement security firewalls may correspond to firewalls that process traffic exchanged between the source and destination nodes. The method may be implemented on, for example, network 100 (FIG. 1A and FIG. 1B), using policy server 200 (FIG. 2).

[0077] In step 410, a new adjacent node is located from known node. Initially, the known node is source node 102 (FIG. 1A and FIG. 1B), and the first new adjacent node is adjacent to a source node. As the method forward traverses to other nodes, each new adjacent node is located from a previously located node.

[0078] In step 420, a path is identified for each new adjacent node. The path corresponds to a sequence of adjacent nodes. An identified path may or may not be a communication path extending between the source node and the destination node.

[0079] In step 425, a determination is made as to whether the adjacent node located in step 410 is the destination node. If the determination in step 425 is that the adjacent node is the destination node then step 430 provides that each node identified as part of the path in step 420 is also designated as being part of a path closure set.

[0080] In step 435, a determination is made as to whether there are any flagged nodes. As will be described, flag nodes are nodes in looping paths that are not yet determined as being part of the path closure set. If there are no flagged nodes, the method proceeds with step 440.

[0081] In step 440, a determination that follows step 430. In step 440, a determination is made as to whether any nodes are left to be located in the network topology. If the determination is positive, step 445 backtracks to identify a previous node.

-18-

50325-0626 (Seq. No. 4895)

[0082] Step 450 determines whether there is a new adjacent node located at the previous node. If the determination in step 450 is negative, then step 455 provides that the previous node is removed from the path. Step 445 is repeated, and another previous node is identified.

[0083] If the determination in step 450 is positive, then step 410 is repeated for the new adjacent node.

[0084] If the determination in step 425 is that the adjacent node is not the destination node, then step 460 makes the determination as to whether the adjacent node is a loop closure node. A loop closure node may correspond to a node in a sequence of adjacent nodes that was previously located and is part of the same path. If the determination in step 460 is negative, then step 410 is repeated. The node for step 440 becomes a previously located node, and a new node is located that is adjacent to the previously located node.

[0085] If the determination in step 460 is positive, then step 465 makes a determination as to whether any nodes in the path, other than the node identified in step 460, are part of the path closure set. This determination corresponds to the second condition. For example, if the destination node is found before the loop path, then step 465 provides that nodes in the loop path are checked to determine if any nodes in the loop path, other than the path closure node, are part of the path closure set.

[0086] If the determination in step 465 is positive, then step 470 provides that each node in the path is identified as part of the path closure set. This means that a node other than the path closure node is part of the path closure set. Therefore, each node in the loop path is part of the closure set.

[0087] Step 475 follows step 470 and makes the determination as to whether there are any adjacent nodes to the path closure node found in step 460. If there are adjacent nodes, step 410 is repeated. If there are no adjacent nodes, then one embodiment provides that the

-19-

method backtracks to a previous node. Accordingly, the method continues by performing step 445.

[0088]   If the determination in step 465 is negative, then step 480 provides that the first node located after the path closure node was located the first time is flagged. If this node is subsequently determined to be part of the path closure set, then all nodes in the loop path identified by step 465 are made part of the path closure set. The method proceeds to step 475, to determine whether there are any adjacent nodes to the path closure node found in step 460.

[0089]   If step 445 determines there are flagged nodes, then step 485 determines if any nodes determined to be part of the path closure set include one of the flagged nodes. If the determination is positive, then step 485 determines that each node in the loop path corresponding to the flagged nodes is made part of the path closure set.

[0090]   The method repeats step 440 until no nodes remain in the network topology to be checked. Once all nodes are checked, the method is done.

[0091]   Among other advantages, a method such as detailed by steps 410-485 may be completed in a duration that is linear, or directly proportional, to the number of links between all nodes in the network topology. Thus, the determination of enforcement firewalls and security devices does not become exponentially more extensive. This is in contrast to the "brute-force" method, where every conceivable path between source and destination nodes is determined, including looping paths that have only the path closure node as part of the path closure set.

4.0 HARDWARE OVERVIEW

[0092]   FIG. 5 is a block diagram that illustrates a computer system 500 upon which an embodiment of the invention may be implemented. Computer system 500 includes a bus 502 or other communication mechanism for communicating information, and a processor 504

5   coupled with bus 502 for processing information. Computer system 500 also includes a main memory 506, such as a random access memory ("RAM") or other dynamic storage device, coupled to bus 502 for storing information and instructions to be executed by processor 504. Main memory 506 also may be used for storing temporary variables or other intermediate information during execution of instructions to be executed by processor 504. Computer

10   system 500 further includes a read only memory ("ROM") 508 or other static storage device coupled to bus 502 for storing static information and instructions for processor 504. A storage device 510, such as a magnetic disk or optical disk, is provided and coupled to bus 502 for storing information and instructions.

[0093]   Computer system 500 may be coupled via bus 502 to a display 512, such as a

15   cathode ray tube ("CRT"), for displaying information to a computer user. An input device 514, including alphanumeric and other keys, is coupled to bus 502 for communicating information and command selections to processor 504. Another type of user input device is cursor control 516, such as a mouse, trackball, stylus, or cursor direction keys for communicating direction information and command selections to processor 504 and for

20   controlling cursor movement on display 512. This input device typically has two degrees of freedom in two axes, a first axis (e.g., x) and a second axis (e.g., y), that allows the device to specify positions in a plane.

[0094]   The invention is related to the use of computer system 500 for determining enforcement firewalls in a network. According to one embodiment of the invention,

-21-

determining enforcement firewalls in a network is provided by computer system 500 in response to processor 504 executing one or more sequences of one or more instructions contained in main memory 506. Such instructions may be read into main memory 506 from another computer-readable medium, such as storage device 510. Execution of the sequences

5 of instructions contained in main memory 506 causes processor 504 to perform the process steps described herein. In alternative embodiments, hard-wired circuitry may be used in place of or in combination with software instructions to implement the invention. Thus, embodiments of the invention are not limited to any specific combination of hardware circuitry and software.

10 [0095] The term "computer-readable medium" as used herein refers to any medium that participates in providing instructions to processor 504 for execution. Such a medium may take many forms, including but not limited to, non-volatile media, volatile media, and transmission media. Non-volatile media includes, for example, optical or magnetic disks, such as storage device 510. Volatile media includes dynamic memory, such as main memory

15 506. Transmission media includes coaxial cables, copper wire and fiber optics, including the wires that comprise bus 502. Transmission media can also take the form of acoustic or light waves, such as those generated during radio-wave and infra-red data communications.

[0096] Common forms of computer-readable media include, for example, a floppy disk, a flexible disk, hard disk, magnetic tape, or any other magnetic medium, a CD-ROM, any other

20 optical medium, punchcards, papertape, any other physical medium with patterns of holes, a RAM, a PROM, and EPROM, a FLASH-EPROM, any other memory chip or cartridge, a carrier wave as described hereinafter, or any other medium from which a computer can read.

[0097] Various forms of computer readable media may be involved in carrying one or more sequences of one or more instructions to processor 504 for execution. For example, the

-22-

instructions may initially be carried on a magnetic disk of a remote computer. The remote computer can load the instructions into its dynamic memory and send the instructions over a telephone line using a modem. A modem local to computer system 500 can receive the data on the telephone line and use an infra-red transmitter to convert the data to an infra-red

5   signal. An infra-red detector can receive the data carried in the infra-red signal and appropriate circuitry can place the data on bus 502. Bus 502 carries the data to main memory 506, from which processor 504 retrieves and executes the instructions. The instructions received by main memory 506 may optionally be stored on storage device 510 either before or after execution by processor 504.

10   [0098]   Computer system 500 also includes a communication interface 518 coupled to bus 502. Communication interface 518 provides a two-way data communication coupling to a network link 520 that is connected to a local network 522. For example, communication interface 518 may be an integrated services digital network ("ISDN") card or a modem to provide a data communication connection to a corresponding type of telephone line. As

15   another example, communication interface 518 may be a local area network ("LAN") card to provide a data communication connection to a compatible LAN. Wireless links may also be implemented. In any such implementation, communication interface 518 sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information.

20   [0099]   Network link 520 typically provides data communication through one or more networks to other data devices. For example, network link 520 may provide a connection through local network 522 to a host computer 524 or to data equipment operated by an Internet Service Provider ("ISP") 526. ISP 526 in turn provides data communication services through the world wide packet data communication network now commonly referred to as

-23-

the "Internet" 528. Local network 522 and Internet 528 both use electrical, electromagnetic or optical signals that carry digital data streams. The signals through the various networks and the signals on network link 520 and through communication interface 518, which carry the digital data to and from computer system 500, are exemplary forms of carrier waves

5   transporting the information.

[00100]   Computer system 500 can send messages and receive data, including program code, through the network(s), network link 520 and communication interface 518. In the Internet example, a server 530 might transmit a requested code for an application program through Internet 528, ISP 526, local network 522 and communication interface 518. In

10   accordance with the invention, one such downloaded application provides for determining enforcement security devices on a network, as described herein.

[00101]   The received code may be executed by processor 504 as it is received, and/or stored in storage device 510, or other non-volatile storage for later execution. In this manner, computer system 500 may obtain application code in the form of a carrier wave.

15   5.0   EXTENSIONS AND ALTERNATIVES

[00102]   In the foregoing specification, the invention has been described with reference to specific embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. The specification and drawings are, accordingly, to be regarded in an illustrative

20   rather than a restrictive sense.